2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Shoreline extraction from the fusion of LiDAR DEM data and aerial images using mutual information and genetic algrithms

Amr Yousef and Khan Iftekharuddin

***Abstract*—As sea level rises and coastal populations continue to grow, there is an increased demand for understanding the accurate position of the shorelines. The automatic extraction of shorelines utilizing the digital elevation models (DEMs) obtained from light detection and ranging (LiDAR), aerial images and multi-spectral images has become very promising. In this paper, we propose a new algorithm that can effectively extract shorelines from fused LiDAR DEMs with aerial images depending on the availability of training data. The LiDAR data and the aerial image are fused together by maximizing the mutual information using the genetic algorithm. The extraction of shoreline is obtained by segmenting the fused data into water and land by means of the support vector machines classifier. Compared with other relevant techniques in literature, the proposed method offers better accuracy in shoreline extraction.**

***Keywords*: Remote sensing, LiDAR, DEM, Support vector machines, Genetic algorithms, Mutual information**

I. Introduction

Shoreline is a spatial varying separation between water and land [1]. Its erosion and accretion play an essential role in coastal protection design, sea level rise monitoring, historical rate of change quantification and coastal zones developments policies formulation [2]. The mapping of a shoreline is based on selection of good features that can robustly handle both temporal and spatial variations of its positions within the available data sources. Several data sources such as: historical land-based photographs, coastal maps and charts, aerial images, beach surveys, multispectral/hyperspectral images, LiDAR DEM data and microwave sensors can be used to extract the shoreline locations.

Aerial images typically have broad spatial coverage but their temporal coverage is limited by the acquisition time. The Hyperspectral images provide broad spectral coverage but they are limited due to low pixel resolution. LiDAR data can cover a wide spread of regions in short periods of time and provide high resolution digital elevation models that are both accurate and cost effective. Unlike the aerial photographs that utilize the wet-dry boundary as a shoreline proxy which is affected by tidal effects and wave movements, LiDAR DEM data can be used to extract true shoreline positions as they are referenced to tidal datum gauge measurements [3]. Additionally, a single LiDAR DEM data acquired at the lowest tide can be used to extract shorelines referenced to different datums such as mean high water (MHW), mean low water (MLW), mean higher high water (MHHW), mean lower low water (MLLW) and mean surface water (MSW) which can not be achieved through aerial images or hyperspectral images [3].

There have been many attempts to effectively automate the shoreline extraction. Descombes et al. [4] presented an edge detection algorithm for extracting shorelines from a satellite Synthetic Aperture Radar (SAR) image. Ryan et al. [5] proposed an image segmentation approach that was tested on scanned U.S. geological survey (USGS) aerial photographs. Mason and Davenport [6] employed an edge detection method based on a coarse-fine resolution processing strategy and applied it to satellite SAR images. Liu and Jezek [7] developed an automated shoreline extraction method based on a locally adaptive thresholding algorithm that was used on both optical and radar images. The fusion of multi-modal remotely sensed data to extract shoreline position has been reported in literature. Wu et al. [8] extracted water features from aerial images fused with LiDAR data. They constructed a triangular irregular network (TIN) from LiDAR points by means of quad-edge based incremental insertion algorithm. Rough water features are obtained by analyzing the area of the generated TIN. They utilized the mean shift segmentation algorithm [9], [10] to obtain a finer classification of the rough water areas. Lee et al. [11], [12] proposed a method for shoreline extraction from integrated LiDAR point cloud data and aerial orthophotos using mean shift segmentation. They trained the mean shift segmentation on the LiDAR elevations only to select the best bandwidth parameter that maximizes the total true positive classification rate. Since the mean shift segmentation serves as a segmentation rather than a classification algorithm, they manually digitize a partial shoreline segment to serve as a ground truth that is used in classifying the segmented land and water regions. Their classification is based on the homogeneous nature of the elevation and color distribution of a water surface which is expected to give inaccurate results in areas where the water may submerge lands.

In this paper, we develop a new approach to extract shorelines from the fused liDAR DEMs and their corresponding coverage of aerial images with the support of training data. The multi-modal data are fused by maximizing their mutual information by means of the genetic algorithm [13]. Additionally, support vector machines classify the fused data and segment it into land and water pixels. This approach can work without reference to a tidal datum and can extract the MHW shoreline but if tidal datum exist, it can be used

Amr Yousef with the Department of Enginering Mathemtics and Physics, Alexandria University, Egypt(email: {amr.yousef}@alexu.edu.eg)
Khan Iftekharuddin with the Department of Electrical and Computer Engineering, Old Dominion University (email: {kiftekha}@odu.edu).

to extract other shorelines (MLW, MLLW, etc.) as well. We assess the accuracy of the proposed approaches by comparing to a ground truth. In addition, we compare our approach with recent shoreline extraction methods such as Lee et al. [11].

The rest of this paper is organized as follows: In section 2, we describe the LiDAR DEM. In section 3, we describe the Mutual information-based data fusion using genetic algorithm. Section 4 demonstrates the SVM based approach while section 5 discusses the results introduced in this work and section 6 concludes the overall study.

II. LIDAR DEMs

The LiDAR data is a cloud of irregularly distributed points with X, Y, Z coordinates. Generally, to create digital elevation models, the LiDAR point cloud is processed in several steps. The cloud data is filtered and interpolated into a grid with the required spatial resolution. If one or more LiDAR points are found in a grid cell, the median Z for those points is taken as the value for the grid cell. For any grid cell where no LiDAR points are found, a Z value is determined using an inverse distance weighted interpolation with the surrounding neighboring points. The horizontal co-ordinates of the LiDAR points are referenced to the North American Datum of 1983 (NAD83). In our study, we downloaded the DEM data from the NOAA Coastal Services Center (http://csc.noaa.gov/digitalcoast/data/coastallidar/index.html) with point density of 0.1 to 8 pts/meter^2 and elevations accuracy of 30 centimeters at 95% confidence interval.

In the processed DEM files, there are segments with "not a number" (NAN) values in LiDAR data acquisition which will affect the subsequent steps of automatic shoreline extracting. Accordingly, we use locally weighted scatterplot smoothing (Lowess in Cleveland [14]) nonparametric regression method to estimate the LiDAR data of the NAN elevation regions based on their neighborhoods. We limit Lowess's method to work on a window that is four times larger than the NAN region. The window is centered around the NAN region. The advantage of using Lowess's method is that it doesn't require a specific model to fit all the data and is flexible enough to handle complex models within the sample. We used a tri-cube weight function as defined by:

$$\omega_i = \left(1 - \left|\frac{x - x_i}{d(x)}\right|^3\right)^3 \tag{1}$$

where x is the predictor value associated with the response value to be smoothed, x_i are the nearest neighbors of x as defined by the span, and $d(x)$ is the distance along the abscissa from x to the most distant predictor value within the span. Close inspection shows that the NAN data are located in water and land bodies and away from the coast. Consequently, use of regression method will not affect accuracy of the extracted shorelines.

III. MUTUAL INFORMATION-BASED DATA FUSION USING GENETIC ALGORITHM

Both aerial images and the corresponding DEM data should be referenced to a common grid and have the same spatial resolution. The liDAR DEMs and their corresponding aerial coverages are fused by maximizing their mutual information using a genetic algorithm. Mutual information combines the data, where better classification is usually obtained from diverse data. So, mutual information is used to in order to maximize superposition of the information contained in multi-modal data and hence the discrimination rate can be higher than using individual data sources.

A. Genetic algorithms

Genetic Algorithms are non-linear optimization techniques that finds an optimum solution by defining a cost function controlled by a set of parameters. To obtain an optimum solution, GA uses multiple search paths instead of doing a regular grid search which in turn will reduce time and space. Continuous genetic algorithms (CGA) define the solution in terms of real numbers and they involve 6 elements [13]: (1) cost function; (2) initial population; (3)natural selection; (4) mating; (5) mutation; and (6) next generation.

The Cost function is minimized with respect to floating point variables or parameters known as chromosomes. Each chromosome is represented with an N-dimensional vector $P = [p_1, p_2, \ldots, p_N]$. Since the GAs are search techniques, there should be some constraints on the defined variables.

To start the CGA, an initial population matrix of dimension $N_{pop} \times N$ is defined where N_{pop} is the population size. Each chromosome is evaluated by the cost function. The chromosomes are ranked according to their costs and only strong chromosomes are kept for the mating step while the others are discarded.

There are many approaches for doing the mating between the selected chromosomes. The simplest method is to randomly select a crossover point and all the genes to the right of the selected point are swapped. Since the CGA may converge to non global minimum, mutations or changes in the variables are introduced to void the stuck at a local minimum.

The initial population is then ranked and the top chromosomes will be selected for the next population while the bottom ranked chromosomes are replaced with the initially discarded chromosomes. The process continues until the CGA converges to a global minimum [13].

B. Mutual Information

The entropy of random variable is the amount of information required to represent that variable. For two random variables X and Y with joint probability distribution $p(x, y)$ and their marginal distributions are $p(x)$ and $p(y)$ respectively, the mutual information $I(X;Y)$ is the relative entropy between the joint distribution and the product of their marginal distributions as given by:

$$I(X;Y)=\sum_x\sum_y p(x,y)\log\frac{p(x,y)}{p(x)p(y)} \tag{2}$$

The mutual information can be written as:

$$\begin{aligned} I(X;Y) &= \sum_x\sum_y p(x,y)\log\frac{p(x|y)}{p(x)} \\ &= -\sum_x p(x)\log p(x) \\ &+ \sum_x\sum_y p(x,y)logp(x|y) \\ &= H(X)-H(X|Y) \\ &= H(Y)-H(Y|X) \end{aligned} \tag{3}$$

where $H(X)$ is the entropy of the random variable X and $H(X|Y)$ is the entropy of X with the knowledge of Y. It can be seen that the mutual information represents the reduction in the information of one of the random variables with the knowledge of the other one.

C. Fusion Cost Function

Suppose that the aerial image is $A(x,y)$ and its corresponding LiDAR DEM is $L(x,y)$, then the fused image $F(x,y)$ which will also represent the GA cost function is defined by:

$$F(x,y)=C_aA(x,y)+C_lL(x,y) \tag{4}$$

where C_a and C_l are the corresponding coefficients required to maximize the total mutual information between the fused image and both the aerial image and the liDAR DEM. The GA should find the optimum values of both C_a and C_l to maximize the total mutual information I defined by

$$I=I(F;A)+I(F;L) \tag{5}$$

where $I(F;A)$ and $I(F;L)$ are the mutual information between the fused image and both the aerial and liDAR DEM respectively.

According to Equation 3, maximizing the mutual information between two random variables may result in a loss of the information symmetry. In image fusion, this loss can be evaluated by measuring the fusion symmetry (FS) as given by [15]

$$FS=2-\left|\frac{I(F;A)}{I}-0.5\right| \tag{6}$$

IV. Shoreline extraction using SVM classifier

We propose an efficient and accurate method to extract the shoreline from the fusion of different remotely sensed data sources including: LiDAR data, aerial images and multi-spectral images. Our approach uses both aerial images and DEM data but it can be easily adjusted to handle other data sources. In our proposed algorithm, we use the SVM classifier due to its high accuracy, speed and minimal user interactive needs [16].

A. Support Vector Machines

1) SVM classifier: In this section we give a short review on the support vector machines. Let $(\mathbf{x}_i,y_i)_{1\leq i\leq N}$ be the set of training samples, where $\mathbf{x}\in\Re^d$, d is the dimension of the feature space and $y_i\in\{-1,1\}$ is the class labels. If the training samples are linearly separable, and there exists a weight vector $\mathbf{w}$ then the classification criteria (CC) is given by [17]:

$$\text{CC}=\begin{cases} y_i=-1 & \mathbf{w}\cdot\mathbf{x}_i\geq 1 \\ y_i=-1 & \mathbf{w}\cdot\mathbf{x}_i\leq -1\end{cases} \tag{7}$$

An optimum separating hyperplane can be found by minimizing the squared norm of the hyperplane as given by

$$\text{Minimize}\quad \Phi(\mathbf{w})=\frac{1}{2}\left\|\mathbf{w}\right\|^2\quad \text{constrained with } y_i(\mathbf{w}^T\mathbf{x_i}+b\geq 1) \tag{8}$$

where b is defined as bias . The quantity $\left\|\mathbf{w}\right\|^2$ is convex and can be minimized using Lagrange multipliers. The optimization problem reduces to:

$$W(\alpha)=\sum_{i=1}^N\alpha_i-\frac{1}{2}\sum_{i,j=1}^N\alpha_i\alpha_jy_iy_j\mathbf{x_i}\cdot\mathbf{x}_j \tag{9}$$

After solving for the Lagrange multipliers, the optimum separating hyperplane is given by:

$$\mathbf{w}_o=\sum_{i=1}^N\alpha y_i\mathbf{x}_i \tag{10}$$

and the decision criterion for the samples under test is given by

$$f(\mathbf{x})=\text{sgn}\left(\mathbf{w}_o\cdot\mathbf{x}+b\right) \tag{11}$$

If the data contain misclassified samples and are not linearly separable then the SVM classifier may not find any separating plane. Such a problem can be solved using positive slack variables ξ_i and the new classification scheme is given by [18]:

$$\text{CC}=\begin{cases} y_i=1-\xi_i & \mathbf{w}\cdot\mathbf{x}_i\geq 1 \\ y_i=-1+\xi_i & \mathbf{w}\cdot\mathbf{x}_i\leq -1\end{cases} \tag{12}$$

and the new minimization problem becomes:

$$\text{Minimize}\left\{\frac{1}{2}\left\|\mathbf{w}\right\|^2+D\sum_{i=1}^N\xi_i\right. \tag{13}$$

For misclassified instance to occur, the value ξ_i must be greater than 1 and hence $\sum_i$ is an upper bound for the training errors. The term D is used to compensate for the misclassified instances and can be determined using SVM training. In many realistic cases, there will not be a hyperplane that can classify non-separable instances unless they are mapped to higher dimensional space where the training of the SVM should take place. If $\mathbf{x}_i$ is replaced by its mapping

$\phi(\mathbf{x_i})$ then equation 9 becomes:

$$W(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j \phi(\mathbf{x_i}) \cdot \phi(\mathbf{x_j}) \tag{14}$$

The training algorithm thus, will depend upon the dot product of the mappings rather than depending on $\phi(\mathbf{x})$ themselves. This can be achieved through a predefined kernel $K(\mathbf{x}_i, \mathbf{x}_j)$ such that $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x_i}) \cdot \phi(\mathbf{x_j})$ and consequently, $\phi(\mathbf{x_i})$ is not required to be known explicitly [19]. Once the kernel is chosen the optimization problem is solved then by minimizing:

$$W(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \tag{15}$$

and the classifying criterion becomes:

$$f(\mathbf{x}) = \mathrm{sgn}\left(\sum_{i}^{N} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b \right). \tag{16}$$

2) SVM training: The aerial images are fused on the corresponding LiDAR DEM data. To reduce the computational complexity and memory requirements required for the training and the classification steps, we split the raster grids into non-overlapping partitions. Every partition will be trained and tested independently from the other partitions. Since every partition is self trained and classified, parallel computations can be used to accelerate the whole process. The training is done by means of 10-fold cross validation to select the best kernel with its optimal parameters that can achieve minimal classification error rate. Genton [20] has defined several kernels that can be used in SVM classifier but he did not specify the limitations of each kernel and to what kind of problems it can be applied. Experimentally, we found that the best results can be obtained by a radial basis kernel defined by:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right),\ \gamma > 0 \tag{17}$$

We conducted a grid search on the parameters γ, C that can achieve best by means of 10-folds cross validation on the training samples. The experimental value for both γ and C is 1. Once the classifier is trained the resultant model is applied to the test data. The selection of the training instances does not require much experience of the user and it can be done manually by selecting a number of samples that represent water and land regions. The advantage of this step is that the user can either train the classifier to extract the MHW shoreline or any other shoreline that is referenced to other datums. If a tidal datum is used, the DEM data are segmented based on this threshold and then the user has to select the water samples that fall below the threshold or the land pixels that are above the threshold. After the training samples have been selected, the SVM classifier with radial basis kernel is applied to the training data and tested using 10-folds cross validations to select its optimal parameters.

3) Shoreline Extraction: Once the SVM classifier has been trained, it should be applied to the whole data to classify it into water and land regions. Then a small area removal is carried to eliminate the unwanted areas from the classified image. Finally, The boundary between water and land is extracted and smoothed using a Guassian kernel.

V. RESULTS AND ANALYSIS

We downloaded both LiDAR DEM and the corresponding aerial images from NOAA Coastal Services Center (http://csc.noaa.gov/digitalcoast/data/coastallidar/index.html). The dates of data acquisition are 03/11/08-03/14/08. Both the DEM files and the tidal datums are referenced to NAVD88. The LiDAR data were collected by the NOAA National Geodetic Survey Remote Sensing Division using a Riegl Q680i-D system and they were in universal transverse mercator (UTM), Zone 18 coordinates. The given example extends from -74.98 W to -74.38 E and from 39.4 N to 38.91 S. Its horizontal accuracy is 1 m while its vertical accuracies are 0.15 m. The DEM is a raster grid of elevation values with 2074 columns and 2076 rows. The DEM files are all horizontally and vertically referenced with respect to NAD83 and NAVD88 respectively. We compare the proposed approach against Lee et al. [11] approach. Simulations are done using MATLAB 7.8 Release 2009a program on on OPTIPLEX 780 (Intel(R) Core (TM)2 Quad 2.66 GHz CPU with 8.00 GB.

The evolution of the GA is depicted in Figure 1. The initial population size is 1003 and is set to a double vector. The first and second columns in population matrix represent aerial image and LiDAR DEM while the third one represents the mutual information. The mutation function is set to Gaussian. In each evolution of the GA, only 30 rows are kept for mating and the others are neglected. The mutation rate is set to 0.20, hence the total number of mutated variables is 60. The GA stops at the 51$^{\text{th}}$ iteration. The optimum solution is achieved at the 30$^{\text{th}}$ iteration and remains unchanged till the 51$^{\text{th}}$ evolution. The fusion coefficients are $C_a = 0.9968$ and $C_l = 0.0042$. The results of the multi-modal data in addition to the results of the image fusion are shown in Figure 3. Although the LiDAR fusion coefficient is small compared to the optical image one, the accuracy assessment shows an improved performance in terms of the extraction error. Once the SVM classifier has been trained, the network is applied to the whole data to classify it into water and land regions. Then a small area removal is done to eliminate the unwanted areas from the classified image. Finally, the boundary between water and land is extracted and smoothed using a Guassian kernel.

The extracted shoreline is depicted in Figure 4. Visually, it can be seen from Figure 4(c) and 4(d) that the SVM approach is closer to the ground truth and is better than the approach proposed by Lee et al.[11]

A. Accuracy assessments

We assess the accuracy of the proposed approach by comparing the extracted shorelines against manually extracted ground truth. The NOAA website does not provide ground truth data with matching dataset with exact date, time and spatial resolution information. Accordingly, we traced the MHW shoreline manually and used it as a ground truth. We set a number of transect along the ground truth to calculate the error difference between it and the extracted shorelines. In this assessment, we used example 1 and the transects can be seen as small black segments across the shoreline. We plot error difference between the extracted shorelines and the ground truth at every transects as shown in Figure 2. The average of errors are 2.37 m and and 4.92 m for the SVM, and Lee et al. approaches respectively. The accuracy of our proposed approach is better than the other approach. In a recent work by the authors [21], they utilized the same dataset and the average error was 2.81 m. In their approach, they utilized SVM classifier on both aerial images and LiDAR data without fusion.

VI. Conclusions

In this paper we presented an efficient approach to extract shorelines from remotely sensed data depending on the available data sources and training data. It's assumed that both the aerial images and LiDAR DEM have the same spatial coveraage, the same spatial resolution and georeferenced to a common grid. The proposed approach fuses the multimodal data by maximizing the mutual information between the the fused image and both the aerial image and the liDAR DEM. Then, the fused data are segemnted into land and water by means of SVM classifier. The visual quality assessment of the extracted shoreline is depicted by superimposing the extracted shorelines on their corresponding aerial images and comparing them to a ground truth. It was shown that the SVM approach yields better performance and accuracy than the approach proposed by Lee et al. [11]. Our SVM based approach has an average shoreline position error of 2.37 m while Lee et al. approach has an average error of 4.92 m.

References

[1] E. H. Boak and I. L. Turner, "Shoreline definition and detection: A review," *Journal of Coastal Research*, vol. 21, no. 4, pp. 688–703, 2005.

[2] D. Bellomo, M. J. Pajak, and J. Sparks, "Coastal flood hazards and the national flood insurance program," *Journal of Coastal Research*, vol. 28, pp. 21–26, 1999.

[3] H. Liu, D. Sherman, and S. Gu, "Automated extraction of shorelines from airborne light detection and ranging data and accuracy assessment based on monte carlo simulation," *Journal of Coastal Research*, vol. 23, no. 6, pp. 1359–1369, 2007.

[4] X. Descombes, M. Moctezuma, H. Maître, and J.-P. Rudant, "Coastline detection by a markovian segmentation on SAR images," *Signal Processing*, vol. 55, no. 1, pp. 123–132, 1996.

[5] T. Ryan, P. Sementilli, P. Yuen, and B. Hunt, "Extraction of shoreline features by neural nets and image processing," *Photogrammetric Engineering and Remote Sensing;(United States)*, vol. 57, 1991.

[6] D. C. Mason and I. J. Davenport, "Accurate and efficient determination of the shoreline in ERS-1 SAR images," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 34, no. 5, pp. 1243–1253, 1996.

[7] H. Liu and K. C. Jezek, "A complete high-resolution coastline of Antarctica extracted from orthorectified radarsat SAR imagery," *Photogrammetric Engineering and Remote Sensing*, vol. 70, no. 5, pp. 605–616, 2004.

[8] H. Wu, C. Liu, Y. Zhang, and W. Sun, "Water feature extraction from aerial-image fused with airborne lidar data," in *Urban Remote Sensing Event, 2009 Joint*, pp. 1–7, IEEE, 2009.

[9] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *Information Theory, IEEE Transactions on*, vol. 21, no. 1, pp. 32–40, 1975.

[10] Y. Cheng, "Mean shift, mode seeking, and clustering," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 17, no. 8, pp. 790–799, 1995.

[11] I.-C. Lee, B. Wu, and R. Li, "Shoreline extraction from the integration of lidar point cloud data and aerial orthophotos using mean shift segmentation," in *Proceedings of the ASPRS Annual Conference,(Baltimore, MD, 2009)*, 2009.

[12] I.-C. Lee, L. Cheng, and R. Li, "Optimal parameter determination for mean-shift segmentation-based shoreline extraction using lidar data, aerial orthophotos, and satellite imagery," in *ASPRS Conference*, 2010.

[13] R. Haupt and S. Haupt, *Practical Genetic Algorithms*. Wiley, 2004.

[14] W. S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," *Journal of the American Statistical Association*, vol. 74, no. 368, pp. pp. 829–836, 1979.

[15] S. Erkanli, J. Li, and E. Oguslu, *Fusion of Visual and Thermal Images Using Genetic Algorithms*. Old Dominion University, 2011.

[16] S. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *Frontiers in Artificial Intelligence and Applications*, vol. 160, p. 3, 2007.

[17] B. Schölkopf, A. Smola, K.-R. Müller, C. Burges, and V. Vapnik, "Support vector methods in learning and feature extraction," 1998.

[18] K. Veropoulos, C. Campbell, N. Cristianini, *et al.*, "Controlling the sensitivity of support vector machines," in *Proceedings of the International Joint Conference on Artificial Intelligence*, vol. 1999, pp. 55–60, Citeseer, 1999.

[19] B. Schölkopf, C. J. Burges, and A. J. Smola, *Advances in kernel methods: support vector learning*. MIT press, 1998.

[20] M. G. Genton, "Classes of kernels for machine learning: a statistics perspective," *The Journal of Machine Learning Research*, vol. 2, pp. 299–312, 2002.

[21] A. Yousef, K. M. Iftekharuddin, and M. A. Karim, "Shoreline extraction from light detection and ranging digital elevation model data and aerial images," *Optical Engineering*, vol. 53, no. 1, pp. 011006–011006, 2014.

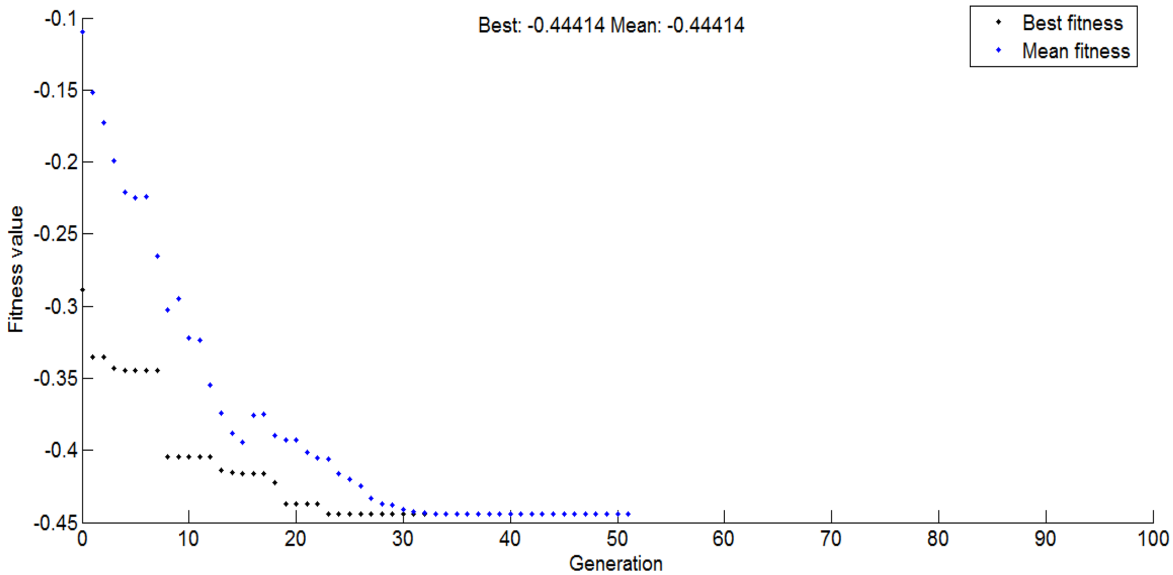


Fig. 1: Genetic algorithm iterations.

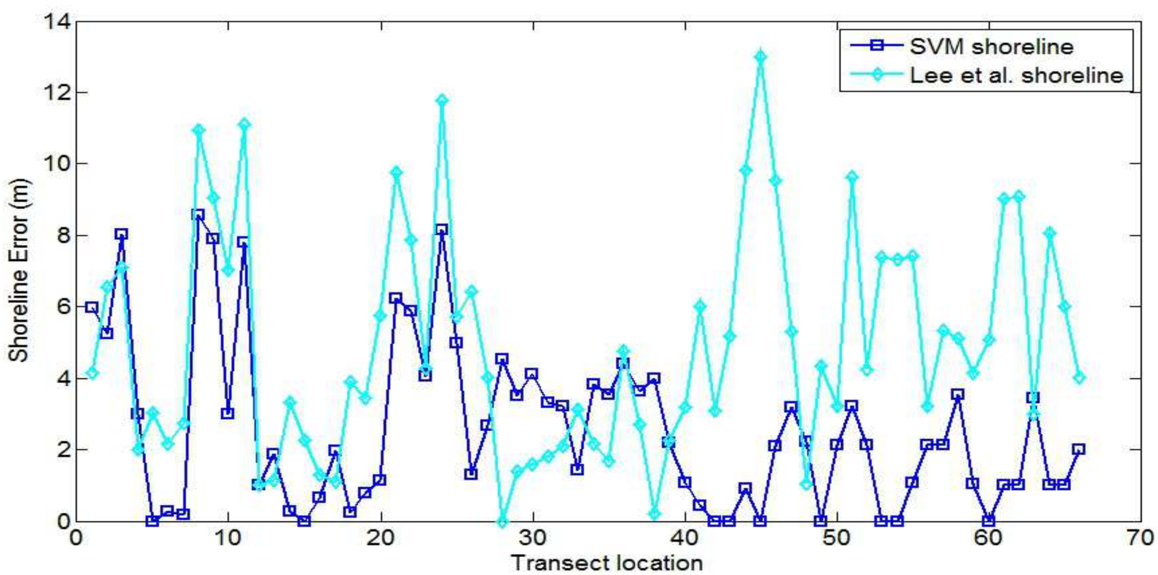
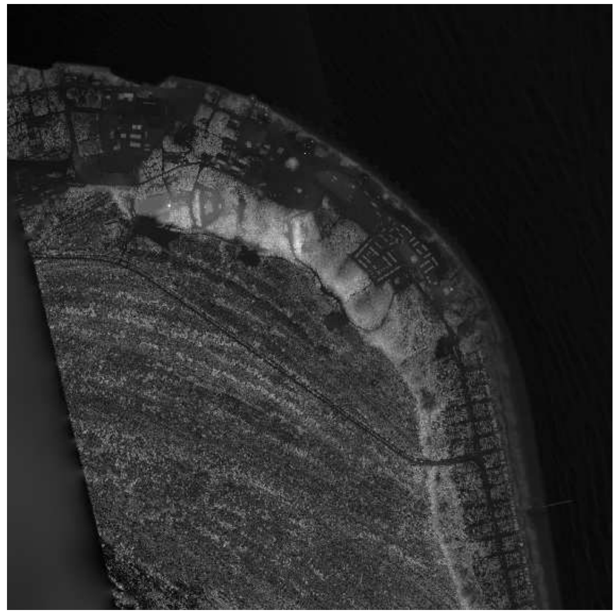


Fig. 2: Absolute error differences in the extracted shorelines from the ground truth at every transect.

(a) Aerial image

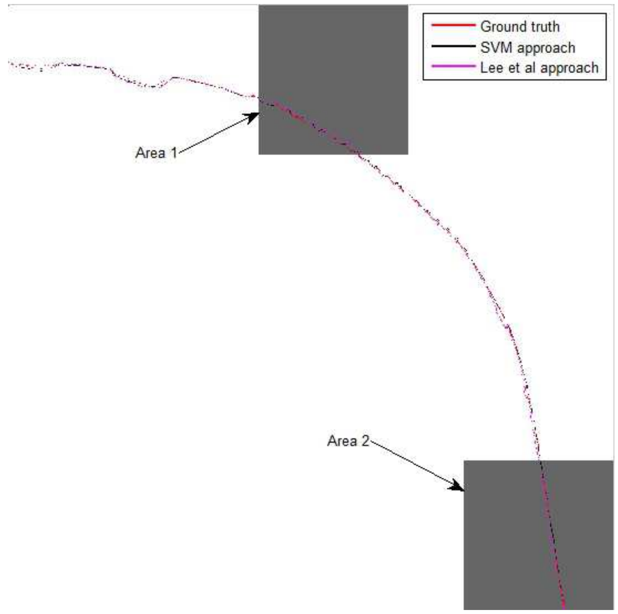


(b) LiDAR DEM

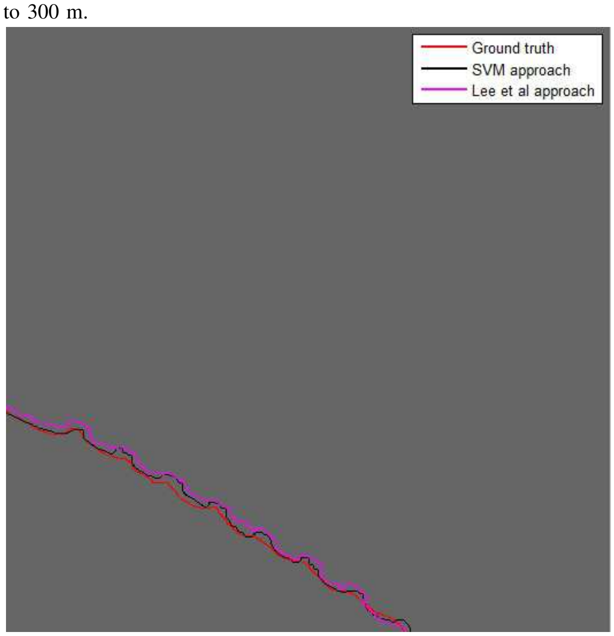
(c) Fused Data

Fig. 3: Gentic algorithm results

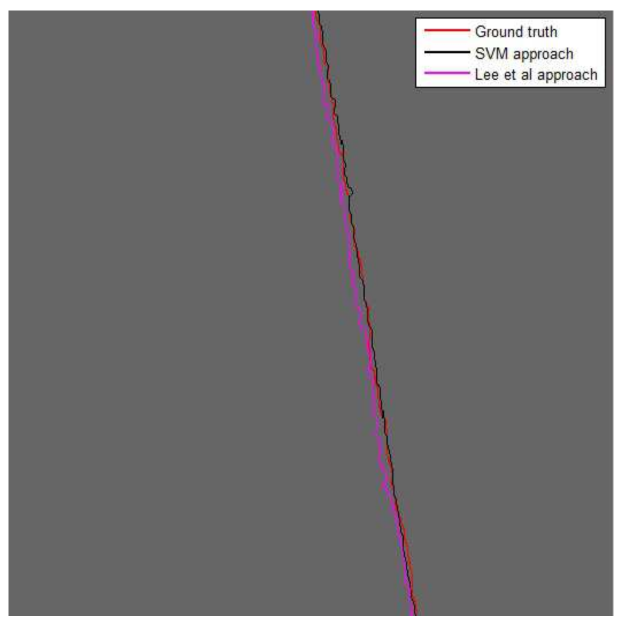
(a) Extracted shoreline superimposed on an aerial image. Scale: 1 cm to 300 m.



(b) Extracted shoreline shapes. 1 cm to 300 m.



(c) Area 1. Scale: 1 cm to 75 m.



(d) Area 2. Scale: 1 cm to 75 m.

Fig. 4: Extracted Shoreline for the coordinates:-74.98W :-74.38E and 39.40N : 38.91S. Comparison between the different approaches: Lee et al. [11], and SVM shoreline. (a) Extracted shoreline superimposed on aerial image; (b) Extracted shoreline profiles; (c) Zoom in area 1 and (d) Zoom in area 2